Group Presentations: Logic Programming with Prolog

Presentation schedule:

Presentation starts: March 31st - May 2nd

Group 1: Introduction to Logic Programming

Goal: Explain the logic programming paradigm and contrast it with imperative programming.

Presentation Content:

- Define programming paradigms and place logic programming within them.
- Explain declarative programming with examples.
- Introduce facts, rules, and queries.
- Walk through a simple family knowledge base.
- Code example:
- parent(john, mary).
- parent(mary, anna).
- grandparent(X, Y) :- parent(X, Z), parent(Z, Y).
- In Class Activities:
- Ask students to predict the result of a query before running it.
- Run the query grandparent(john, anna) and discuss why it succeeds.
- Have students write their own three facts and one rule about a domain such as friends or courses.
- Invite a few students to share their rules and test them live.

Group 2: Installing and Running Prolog

Goal: Guide students through installing and using SWI Prolog to run programs.

Presentation Content:

- Introduce Prolog and its role in logic programming.
- Explain why SWI Prolog is commonly used.
- Provide step by step installation instructions for different operating systems.
- Show how to open the interpreter and run it from terminal or command prompt.
- Explain how to create a .pl file and consult it.

- Example program and query execution.
- In Class Activities:
- Students download and install SWI Prolog on their machines.
- Students open the interpreter and confirm it runs correctly.
- Create a simple file named family.pl and load it.
- Run at least two queries and verify the outputs.
- Troubleshoot installation or execution issues with peers.

Group 3: Rules and Inference

Goal: Demonstrate how Prolog uses rules to perform logical inference.

- Presentation Content:
- Explain the structure of rules using head and body.
- Describe how Prolog evaluates rules.
- Introduce the concept of inference.
- Explain backtracking with a step by step example.
- Provide sample rules and queries.
- In Class Activities:
- Run a query with variables such as grandparent(X, anna).
- Ask students to trace how Prolog finds the answer.
- Modify a rule and observe how results change.
- Have students create one new rule based on existing facts.
- Test multiple queries and interpret results.

Group 4: Lists and Recursion

Goal: Teach how Prolog handles structured data and recursion.

Presentation Content:

- Explain list syntax and notation.
- Describe head and tail decomposition.
- Introduce recursion and how it replaces loops.
- Walk through a recursive example step by step.
- Example code for computing length of a list.
- In Class Activities:
- Run the length predicate on different lists.
- Ask students to manually trace recursive calls.
- Have students write a rule to check if an element exists in a list.
- Test the rule with multiple inputs.
- Discuss common mistakes in recursion.

Group 5: Control and Search in Prolog

Goal: Explain how Prolog explores solutions using its search strategy.

Presentation Content:

- Describe Prolog execution model.
- Explain depth first search.
- Discuss how rule order affects results.
- Introduce the idea of multiple solutions.
- Provide examples where Prolog backtracks.
- In Class Activities:
- Run a query that produces multiple answers.
- Use the semicolon key to explore additional results.
- Change the order of rules and observe differences.
- Have students predict outputs before running queries.
- Discuss efficiency and possible infinite loops.

Group 6: Applications and Mini Project

Goal: Apply logic programming concepts to build a small knowledge based system.

Presentation Content:

- Discuss applications in artificial intelligence and expert systems.
- Explain how knowledge bases are structured.
- Present a simple diagnostic example.
- Show how rules combine to produce conclusions.
- In Class Activities:
- Run queries on the provided example and interpret results.
- Each group creates a small knowledge base with at least five facts.
- Define at least two rules that derive new information.
- Prepare three queries and test them live.
- Present the mini project and explain how the logic works.